

```

#include<stdio.h>
#include<string.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<fcntl.h>
#include<stdlib.h>

main(int argc, char * argv[])

{
    int n, i, maxi, max_fd, k;
    int sock_fd, listen_fd, connfd, client_no;
    int nready, num_q, client[100], chat[100], conn[1000];
    char line[1000], buffer[1000];

    fd_set rset, allset;
    struct sockaddr_in servaddr, cliaddr;

    int len = sizeof(cliaddr);
    bzero(line, sizeof(line));

    client_no = 0;

    if (argc != 2) {
        fprintf(stderr, "Usage: ./server port\n");
        exit(1);
    }

    if ((listen_fd = socket(AF_INET, SOCK_STREAM, 0)) < 0) {
        printf("Cannot create socket\n");
        exit(1);
    }

    bzero((char * ) & servaddr, sizeof(servaddr));

    servaddr.sin_family = AF_INET;
    servaddr.sin_port = htons(atoi(argv[1]));
    servaddr.sin_addr.s_addr = htonl(INADDR_ANY);

    if (bind(listen_fd, (struct sockaddr * ) & servaddr, sizeof(servaddr)) < 0)
    {
        perror("bind failed:");
        exit(1);
    }

    listen(listen_fd, num_q);
    max_fd = listen_fd;

    maxi = -1;

    for (i = 0; i < 100; i++) {
        client[i] = -1;
        chat[i] = -1;
    }
    FD_ZERO( & allset);
    FD_SET(listen_fd, & allset);

    for (;;) {
        rset = allset;
        nready = select(max_fd + 1, & rset, NULL, NULL, NULL);

        if (FD_ISSET(listen_fd, & rset)) {

```

```

        if ((connfd = accept(listen_fd, (struct sockaddr * ) & cliaddr, &
len)) < 0) {
            perror("accept failed");
            exit(1);
        }

        chat[++client_no] = connfd;
        conn[connfd] = client_no;

        k = client_no;
        sprintf(buffer, "Client %d has joined chat\n", k);

        while (k > 0) {
            if (chat[k] > 0)
                n = write(chat[k--], buffer, strlen(buffer));
        }

        for (i = 0; i < 100; i++) {
            if (client[i] < 0) {
                client[i] = connfd;
                break;
            }
        }

        FD_SET(connfd, & allset);
        if (connfd > max_fd) max_fd = connfd;

        if (i > maxi) maxi = i;

        if (--nready <= 0) continue;
    }
    for (i = 0; i <= maxi; i++) {
        bzero(line, sizeof(line));

        if ((sock_fd = client[i]) < 0) continue;

        if (FD_ISSET(sock_fd, & rset)) {
            n = read(sock_fd, line, sizeof(line));
            if (n == 0) {
                close(sock_fd);
                FD_CLR(sock_fd, & allset);
                client[i] = -1;

                bzero(buffer, sizeof(buffer));
                sprintf(buffer, "Client %d has left chat\n", conn[sock_fd]);

                k = client_no;
                while (k > 0) {
                    if (chat[k] > 0)
                        n = write(chat[k--], buffer, strlen(buffer));
                }
            } else {
                if (chat[line[0] - 48] > 0) write(chat[line[0] - 48], line,
strlen(line));
            }

            if (--nready <= 0) break;
        }
    }
}

```